

Research Article

An Improved EKF-SLAM for Mars Surface Exploration

Bo Zheng and Zexu Zhang 

School of Astronautics, Harbin Institute of Technology, 150080 Harbin, China

Correspondence should be addressed to Zexu Zhang; zexuzhang@hit.edu.cn

Received 17 July 2018; Revised 7 November 2018; Accepted 13 November 2018; Published 4 February 2019

Academic Editor: Paolo Gasbarri

Copyright © 2019 Bo Zheng and Zexu Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the traditional EKF-SLAM algorithm, the computational complexity and uncertainty will grow up rapidly with the increase of the feature points and the enlargement of the map coverage. As we know, the computational complexity is proportional to the quadratic of the number of feature points contained in a single filtering process. The approach represented in the paper combines EKF-SLAM with local submaps, which can improve the computational efficiency and reduce the computational complexity. At first, an independent local submap is established for the observed feature points. When the number of feature points contained in the local submap reaches a certain threshold value, the local submap is integrated into the global map. At last, the submap is initialized again. The simulation results show that the approach can reduce the computational complexity effectively and increase the computation speed greatly in the case of maintaining the computational accuracy of the traditional EKF-SLAM algorithm.

1. Introduction

With the development of the manned deep space exploration, Mars as the most suitable planet for human habitation has attracted extensive attention. The traditional flying-around method can no longer satisfy the deeper exploration mission. Therefore, the cruising exploration method of the Mars rover has become an excellent choice [1]. The relative pose information of the Mars rover can be obtained by traditional odometer during the cruising exploration process, which is not accurate on the complex Mars terrain. The navigation accuracy will be improved through the visual aid navigation method in this circumstance, especially SLAM method [2, 3]. The SLAM problem can be described as a problem that you place a robot in an unknown environment with sensors, it can build the incremental maps and achieve its location autonomously during the movement [4].

The initial framework for the navigation of the mobile robot was presented by Dissanayake et al. in the 1980s. Meanwhile, he established the SLAM navigation algorithm for the mobile robot based on Kalman filter [5]. The analysis of the consistency of the EKF-SLAM was presented by Bailey

and Durrant-Whyte [6]. The robocentric map joining method was presented by Castellanos et al. for improving the consistency of EKF-SLAM [7]. The bounding uncertainty of the robocentric local map sequencing algorithm was analyzed by Martinez-Cantin and Castellanos [8]. A lot of landmarks need to be observed and introduced into the state equation by EKF-SLAM, which increases the computational complexity and decreases the running speed of the system [9, 6]. Therefore, the traditional EKF-SLAM algorithm needs to be improved. The computational cost analysis of EKF-SLAM was done by time measurement on substep motion update and measurement update on EKF by considering the total number of landmarks and numerous setting on range observation distance by Samsuri et al. [10], which showed the number of landmarks and the range observation distances were the factors to cause computational cost. A computationally efficient SLAM (CESLAM) algorithm was presented by Hsu et al. [11], where odometer information was considered for updating the robot's pose in particles. When a measurement had a maximum likelihood with the known landmark in the particle, the particle state was updated before updating the landmark estimates. The

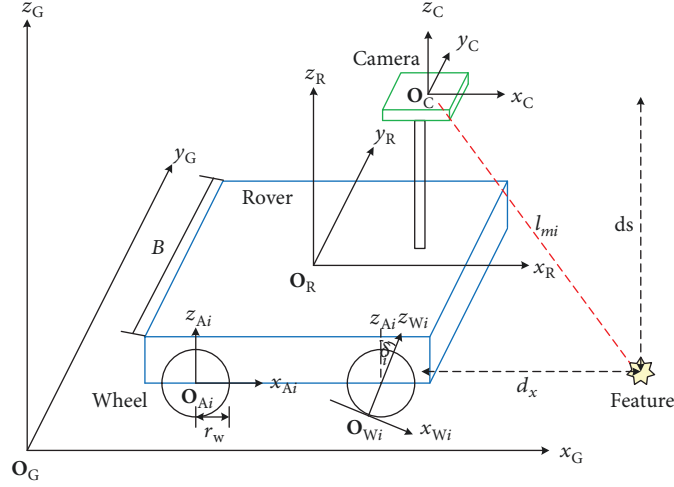


FIGURE 1: Coordinate system of the Mars rover's motion.

algorithm aimed to improve the computational efficiency and estimation accuracy of conventional SLAM algorithm.

An improved EKF-SLAM with local submaps is presented in the paper. Firstly, the feature points observed by binocular camera are grouped; meanwhile, a few submaps are established, which are independent of the global map. And then only the local state variables are updated by the new algorithm in each update process, because the local submap is integrated into the global map only when the area of the submap reaches a certain scale. The global state variables will not be updated until the local submap is integrated into the global map. The main difference between the standard EKF-SLAM algorithm and the algorithm represented in this paper is the introduction of the local submaps, through which the number of feature points used for each filtering update is reduced. Since the computational complexity is proportional to the quadratic of the number of feature points contained in a single filtering process of EKF and the total computational complexity of EKF-SLAM algorithm is proportional to the cube of the final number of feature points in the state variables, the computational complexity is reduced by the improvement. The paper is organized as follows: Section 2 builds the motion model equation of the Mars rover. The EKF-SLAM algorithm is improved based on the local submaps in Section 3. Simulation results and the comparison of the standard EKF-SLAM and the SLAM with submaps are given in Section 4. Finally, the conclusion is drawn in the last section.

2. Motion Model of the Mars Rover

The set of the coordinate system of the Mars rover's motion is given in Figure 1.

As is shown in Figure 1, we denote $\mathbf{O}_G - x_G y_G z_G$ as the inertial coordinates and global coordinates; $\mathbf{O}_R - x_R y_R z_R$ as the body coordinates of the Mars rover, where x_R points to the direction of the rover's motion; $\mathbf{O}_C - x_C y_C z_C$ as the binocular camera coordinates, where x_C points to the vertical direction of the camera lens; $\mathbf{O}_{Ai} - x_{Ai} y_{Ai} z_{Ai}$ as the

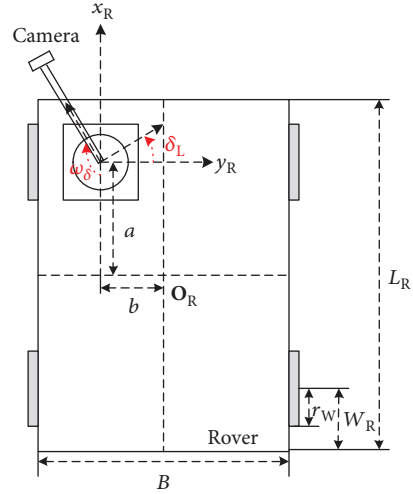


FIGURE 2: The installing model of the binocular camera (vertical view).

coordinates of the rover's i th wheel, which is used to establish the trajectory of the rover's centroid based on the terrain of Mars; and $\mathbf{O}_{Wi} - x_{Wi} y_{Wi} z_{Wi}$ as the coordinates of the contact point for the Mars rover's i th wheel and the planet surface, where x_{Wi} points to the tangential direction of the contact point's surface, which is used to establish the trajectory of the rover's centroid based on the terrain of Mars.

The installing model of the binocular camera is shown in Figure 2 (vertical view) and Figure 3 (lateral view).

The Mars rover's model used in the paper is a four-wheel differential rover with six degrees of freedom. The linear velocity and the angular velocity are controlled input variables, which is decided by angular velocities of differential wheels [12]. The motion equation is as follows:

$$\begin{aligned} \dot{x} &= r_w \left(\frac{\dot{\theta}_L(t) + \dot{\theta}_R(t)}{2} \right), \\ \omega_z &= \frac{r_w}{B} (\dot{\theta}_L(t) - \dot{\theta}_R(t)), \end{aligned} \quad (1)$$

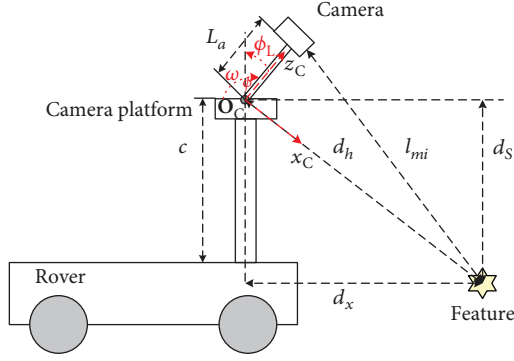


FIGURE 3: The installing model of the binocular camera (lateral view).

where $\dot{\theta}_L(t)$ is the left wheel's velocity, $\dot{\theta}_R(t)$ is the right wheel's velocity, r_w is the wheel's radius, and B is the distance between the left wheel and the right wheel.

The derivative process is omitted; therefore, the kinematic model equation of the Mars rover is given as follows [13]:

$$\begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{z}_R \\ \dot{\phi}_x \\ \dot{\phi}_y \\ \dot{\phi}_z \end{bmatrix} = \begin{bmatrix} \mathbf{C}_R^G & 0 \\ 0 & \mathbf{S}_\omega^G \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix}, \quad (2)$$

where \mathbf{C}_R^G is the rotation matrix; \mathbf{S}_ω^G is the transformation matrix of the angular velocity; $[x_R, y_R, z_R, \phi_x, \phi_y, \phi_z]^T$ denotes pose vectors of the rover, where ϕ_z , ϕ_y , and ϕ_x stand for the yaw angle, pitch angle, and roll angle; \mathbf{v} is the linear velocity of the Mars rover; and $\boldsymbol{\omega}$ is the angular velocity of the Mars rover.

The coordinates of the contact point can be obtained from the coordinates of the Mars rover's wheel [14].

$$\begin{bmatrix} X_{W_i}^{A_i} \\ Y_{W_i}^{A_i} \\ Z_{W_i}^{A_i} \\ 1 \end{bmatrix} = \mathbf{C}_{W_i}^{A_i} \begin{bmatrix} X_{W_i} \\ Y_{W_i} \\ Z_{W_i} \\ 1 \end{bmatrix}, \quad i = 1, 2, 3, 4, \quad (3)$$

where $\mathbf{C}_{W_i}^{A_i}$ is the transformation matrix from the contact point coordinates to the Mars rover's coordinates and $(X_{W_i}, Y_{W_i}, Z_{W_i})$ is the contact point's coordinate.

The binocular camera is fixed on an oblique stick with two degrees of freedom at a certain angle. $u_L = [\omega_\delta, \omega_\phi]$ are the control inputs of the binocular camera, through which a wide range of 3D terrain can be scanned. ω_δ is the rotation angle of the control stick around the platform, and ω_ϕ is

the rotation angle of the control stick around the connecting bolt. ϕ_L is the angle between the control stick and the vertical direction of the platform. The derivative process is omitted; therefore, the pose of the feature point in global coordinates can be described as follows [15]:

$$m_{i_g} = p_R + \mathbf{C}_R^G (\mathbf{C}_H^R \mathbf{C}_C^H m_{i_c} + d), \quad i = 1, \dots, N, \quad (4)$$

where $p_R = [x_R, y_R, z_R]^T$ is the position of the rover in global coordinates, $d = [a, b, c]$ is the position of binocular camera platform in the Mars rover's coordinates, m_{i_c} is the position of the feature points in the camera coordinates, m_{i_g} is the position of the feature points in the global coordinates, \mathbf{C}_R^G is the transformation matrix from the body coordinates to the global coordinates, \mathbf{C}_H^R is the transformation matrix from the platform coordinates to the body coordinates, and \mathbf{C}_C^H is the transformation matrix from the camera coordinates to the platform coordinates.

3. An Improved EKF-SLAM with Local Submaps

3.1. System State Variables. The system state variables of the local submap algorithm include the Mars rover and the feature points in the local coordinates and the feature points in the global coordinates and the coordinate of the local coordinates in the global coordinates, which can be described as follows:

$$\mathbf{X}_{ls}(k) = \begin{bmatrix} {}^G\mathbf{X}_S(k) \\ {}^G\mathbf{X}_1(k) \\ {}^G\mathbf{X}_2(k) \\ \vdots \\ {}^S\mathbf{X}_R(k) \\ {}^S\mathbf{X}_1(k) \\ \vdots \end{bmatrix}, \quad (5)$$

where ${}^G\mathbf{X}_S(k)$ is the pose vector of the local submap in the global map, ${}^G\mathbf{X}_i(k)$ is the position of the feature point in the global map, ${}^S\mathbf{X}_R(k)$ is the pose of the Mars rover in the local submap, ${}^S\mathbf{X}_i(k)$ is the position of the same feature point in the local submap vector ${}^G\mathbf{X}_S(k)$, and k is the discrete time coefficient. The state variables defined above can be simplified as follows:

$$\mathbf{X}_{ls}(k) = \begin{bmatrix} {}^G\mathbf{X}_S(k) \\ {}^G\mathbf{X}_m(k) \\ {}^S\mathbf{X}_R(k) \\ {}^S\mathbf{X}_m(k) \end{bmatrix}, \quad (6)$$

where ${}^G\mathbf{X}_m(k)$ is the feature point in the global map and ${}^S\mathbf{X}_m(k)$ is the feature point in the local submap. The integration process from the submap to the global map is

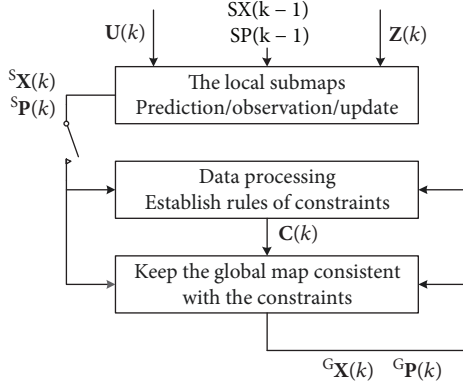


FIGURE 4: The flow chart of the integration process from submap to global map.

shown in Figure 4, where $U(k)$ is the control input, $X(k)$ is the state variable, and $Z(k)$ is the observed variable.

3.2. State Estimation Process. The state covariance matrixes include the covariance matrixes of the global state variables and local state variables and the cross covariance between the global state variables and the local state variables. In this paper, the independence of the global state variables and local state variables is assumed for short intervals, and the short interval independence will be used. The cross covariance between the global state and the local state is zero because

of the independence between the global state variable estimation and the local state variables, which is given as follows:

$$\mathbf{P}_{ls}(k) = \begin{bmatrix} {}^G\mathbf{P}_{ss}(k) & {}^G\mathbf{P}_{ms}(k) & 0 & 0 \\ {}^G\mathbf{P}_{sm}(k) & {}^G\mathbf{P}_{mm}(k) & 0 & 0 \\ 0 & 0 & {}^S\mathbf{P}_{RR}(k) & {}^S\mathbf{P}_{Rm}(k) \\ 0 & 0 & {}^S\mathbf{P}_{mR}(k) & {}^S\mathbf{P}_{mm}(k) \end{bmatrix}, \quad (7)$$

where ${}^G\mathbf{P}_{ss}(k)$ is the covariance matrix of the local submap pose estimation in the global state and ${}^S\mathbf{P}_{RR}(k)$ is the covariance matrix of the Mars rover pose estimation in the local state.

The state predicted value $\mathbf{X}_{ls}(k|k-1)$ is given as follows:

$$\mathbf{X}_{ls}(k|k-1) = \begin{bmatrix} {}^G\mathbf{X}(k-1) \\ f({}^S\mathbf{X}_R(k-1), \mathbf{u}(k)) \\ {}^S\mathbf{X}_m(k-1) \end{bmatrix}. \quad (8)$$

The global estimation is not changed in the local update due to the independence between the local state and the global state, which reduces the computational complexity of the prediction process.

$$\mathbf{P}_{ls}(k|k-1) = \begin{bmatrix} {}^G\mathbf{P}(k-1) & 0 & 0 \\ 0 & \nabla f_X(k) {}^S\mathbf{P}_{RR}(k-1) \nabla f_X^T(k) + \mathbf{Q}(k) & \nabla f_X(k) {}^S\mathbf{P}_{Rm}(k-1) \\ 0 & (\nabla f_X(k) {}^S\mathbf{P}_{Rm}(k-1))^T & {}^S\mathbf{P}_{mm}(k-1) \end{bmatrix}, \quad (9)$$

where $\mathbf{P}_{ls}(k|k-1)$ is the predicted value of the covariance matrix of the state variables.

The state predicted value $\hat{\mathbf{z}}(k|k-1)$ is obtained by the observation model.

$$\hat{\mathbf{z}}(k|k-1) = h({}^S\hat{\mathbf{X}}_R(k-1), {}^S\hat{\mathbf{X}}_i(k-1)). \quad (10)$$

$h({}^S\hat{\mathbf{X}}_R(k-1), {}^S\hat{\mathbf{X}}_i(k-1))$ includes the measure distance, azimuth angle, and rotation angle.

The observation residual and residual covariance matrix of $\hat{\mathbf{X}}$ can be described as follows:

$$\begin{aligned} \mathbf{r}(k) &= \mathbf{z}(k) - \hat{\mathbf{z}}(k|k-1), \\ \mathbf{S}(k) &= \nabla h_X(k) {}^S\mathbf{P}(k|k-1) \nabla h_X^T(k) + \mathbf{Q}_v(k). \end{aligned} \quad (11)$$

Finally, the state update process is given as follows:

$$\begin{aligned} \hat{\mathbf{X}}_{ls}(k) &= \hat{\mathbf{X}}_{ls}(k|k-1) + \mathbf{W}(k)\mathbf{r}(k), \\ \mathbf{P}_{ls}(k) &= \mathbf{P}_{ls}(k|k-1) - \mathbf{W}(k)\mathbf{P}(k)\mathbf{W}^T(k), \\ \mathbf{W}(k) &= \mathbf{P}_{ls}(k|k-1) \nabla h_X^T(k) \mathbf{S}^{-1}(k). \end{aligned} \quad (12)$$

The global state is not updated in the process, because the global state is independent of the local state. The feature points contained in the local state are much less than those contained in the global state. Therefore, the computational complexity is reduced greatly in the update process.

$$\begin{aligned} \mathbf{P}_{ls}(k) &= \begin{bmatrix} {}^G\mathbf{P}(k|k-1) & 0 \\ 0 & {}^S\mathbf{P}(k|k-1) - \mathbf{W}(k)\mathbf{S}(k)\mathbf{W}^T(k) \end{bmatrix}, \\ \mathbf{W}(k) &= {}^S\mathbf{P}(k|k-1) \nabla h_X^T(k) \mathbf{S}^{-1}(k). \end{aligned} \quad (13)$$

3.3. The Independence of Local State Estimation. The local submap established above is composed of the Mars rover's pose state in the local submap and the local submap state. The local submap achieves its initialization by initializing the location of the rover. We hypothesize the initial state matrix is given as follows:

$$\hat{\mathbf{X}}_{ll}(k-1) = \begin{bmatrix} {}^G\hat{\mathbf{X}}_R(k-1) \\ {}^G\hat{\mathbf{X}}_m(k-1) \end{bmatrix}, \quad (14)$$

where ${}^G\hat{\mathbf{X}}_R$ is the estimated value of Mars rover's pose in the global map and ${}^G\hat{\mathbf{X}}_m$ is the estimated value of the feature point position in the global map.

The state covariance matrix is shown as follows:

$$\mathbf{P}_{ll}(k) = \begin{bmatrix} {}^G\mathbf{P}_{RR}(k) & {}^G\mathbf{P}_{Rm}(k) \\ {}^G\mathbf{P}_{mR}(k) & {}^G\mathbf{P}_{mm}(k) \end{bmatrix}, \quad (15)$$

where ${}^G\mathbf{P}_{RR}(k)$ is the covariance matrix of the Mars rover's pose estimation in the global state, ${}^G\mathbf{P}_{mm}(k)$ is the covariance matrix of the feature point position estimation in the global state, and ${}^G\mathbf{P}_{Rm}(k)$ is the covariance matrix of the Mars rover and the feature point in the global state.

A local submap is established at time $k-1$, which takes the Mars rover's position as the center of the submap. Therefore, the Mars rover is located at the origin in the submap whose uncertainty is zero. We hypothesis that the initial state matrix $\hat{\mathbf{X}}_{ll}(k-1)$ is shown as follows:

$$\hat{\mathbf{X}}_{ll}(k-1) = \begin{bmatrix} {}^G\hat{\mathbf{X}}_S(k-1) \\ {}^G\hat{\mathbf{X}}_m(k-1) \\ {}^S\hat{\mathbf{X}}_v(k-1) \end{bmatrix}, \quad (16)$$

$$\mathbf{P}_{ll}(k) = \begin{bmatrix} {}^G\mathbf{P}_{SS}(k) & {}^G\mathbf{P}_{Sm}(k) & 0 \\ {}^G\mathbf{P}_{mS}(k) & {}^G\mathbf{P}_{mm}(k) & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

where ${}^G\mathbf{P}_{SS}(k)$ is the covariance matrix of the local submap pose estimation in the global state and ${}^G\mathbf{P}_{mm}(k)$ is the covariance matrix of the feature point position estimation in the global state.

The noise variance is generated by the local Mars rover's state variables under the modeling error and the observation error during the update process in the local submap. However, this process is independent of global state, which will not affect the global state and covariance matrix. The feature points observed in the new local submap are independent of the global map. Meanwhile, if an independent Mars rover's state estimation is initialized in the local submap, the estimation will maintain its independence to the global map. Therefore, the local state estimation and the global state estimation remain independence.

3.4. The Integration between the Submap and the Global Map. The submap is needed for integration into the global map

when it reaches a certain scale. Then, the submap is rebuilt and initialized again. We hypothesize that $\mathbf{T}(k)$ is the transformation matrix, which can convert the local state to a global state.

$$\begin{aligned} {}^G\hat{\mathbf{X}}_{ls}(k) &= \mathbf{T}_S^G(k) \hat{\mathbf{X}}_{ls}(k), \\ {}^G\mathbf{P}_{cl}(k) &= \nabla \mathbf{T}_S^G(k) \mathbf{P}_{cl}(k) (\nabla \mathbf{T}_S^G(k))^T, \\ {}^G\hat{\mathbf{X}}_{ls}(k) &= \mathbf{T}(k) \begin{bmatrix} {}^G\hat{\mathbf{X}}_S(k) \\ {}^G\hat{\mathbf{X}}_m(k) \\ {}^S\hat{\mathbf{X}}_R(k) \\ {}^S\hat{\mathbf{X}}_m(k) \end{bmatrix} = \begin{bmatrix} {}^G\hat{\mathbf{X}}_S(k) \\ {}^G\hat{\mathbf{X}}_m(k) \\ {}^G\hat{\mathbf{X}}_S(k) \oplus {}^S\hat{\mathbf{X}}_R(k) \\ {}^G\hat{\mathbf{X}}_S(k) \oplus {}^S\hat{\mathbf{X}}_m(k) \end{bmatrix}, \end{aligned} \quad (17)$$

where \oplus denotes the state summation operation in different coordinates including the angular rotation process between different coordinates. We hypothesize that ${}^S\hat{\mathbf{X}}_i(k)$ is the position of a feature point in the local submap.

$${}^S\hat{\mathbf{X}}_i(k) = \begin{bmatrix} {}^Sx_i(k) \\ {}^Sy_i(k) \\ {}^Sz_i(k) \end{bmatrix}. \quad (18)$$

Likewise, we hypothesize that ${}^G\hat{\mathbf{X}}_S(k)$ is the coordinate of the local coordinates in the global coordinates.

$${}^G\hat{\mathbf{X}}_S(k) = \begin{bmatrix} {}^Gx_S(k) \\ {}^Gy_S(k) \\ {}^Gz_S(k) \\ {}^G\phi_{Sx} \\ {}^G\phi_{Sy} \\ {}^G\phi_{Sz} \end{bmatrix}. \quad (19)$$

The global position of the feature point can be obtained as follows:

$${}^G\hat{\mathbf{X}}_{ls}(k) = \begin{bmatrix} {}^G\hat{\mathbf{X}}_S(k) \\ {}^G\hat{\mathbf{X}}_i(k) \end{bmatrix} = \mathbf{T}_S^G(k) \begin{bmatrix} {}^G\hat{\mathbf{X}}_S(k) \\ {}^S\hat{\mathbf{X}}_i(k) \end{bmatrix}, \quad (20)$$

where $\mathbf{T}_S^G(k)$ can be described as follows:

$${}^G\mathbf{X}_R(k) = \mathbf{R}_S^G \begin{bmatrix} {}^G\mathbf{X}_S(k) \\ {}^S\mathbf{X}_R(k) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_6 \\ \mathbf{C}_S^G(k) \\ \mathbf{S}_S^G(k) \end{bmatrix} \begin{bmatrix} {}^G\mathbf{X}_S(k) \\ {}^S\mathbf{X}_R(k) \end{bmatrix}, \quad (21)$$

where $\mathbf{C}_S^G(k) = \mathbf{C}_R^G({}^G\phi_{Sx}, {}^G\phi_{Sy}, {}^G\phi_{Sz})$ which is the transformation matrix from the local coordinates to the global

coordinates. The covariance matrix of the global state variables can be expressed as follows:

$${}^G\mathbf{P}_{ls}(k) = \nabla \mathbf{T}_S^G(k) \begin{bmatrix} {}^G\mathbf{P}_{ss}(k) \\ {}^s\mathbf{P}_{ii}(k) \end{bmatrix} (\nabla \mathbf{T}_S^G(k))^T, \quad (22)$$

where $\nabla \mathbf{T}_S^G(k)$ is the Jacobian matrix of the transformation matrix, which is shown as follows:

$$\nabla \mathbf{T}_S^G(k) = \frac{\partial \mathbf{T}_S^G(k)}{\partial ({}^G\hat{\mathbf{X}}_S(k), {}^s\hat{\mathbf{X}}_i(k))}. \quad (23)$$

We hypothesize that the local pose of the rover is $({}^sx_R(k), {}^sy_R(k), {}^sz_R(k), {}^s\phi_x, {}^s\phi_y, {}^s\phi_z)^T$ and the global pose of the rover is given as follows:

$$\mathbf{T}_S^G(k) = \begin{bmatrix} \mathbf{I}_6 & 0 \\ \mathbf{I}_3 & 0 \end{bmatrix} \mathbf{C}_S^G(k), \quad (24)$$

where $\mathbf{S}_S^G(k) = \mathbf{S}_\omega^G({}^G\phi_{sx}, {}^G\phi_{sy}, {}^G\phi_{sz})$ is the pose transformation matrix. Therefore, the covariance matrix of the Mars rover in the global coordinates can be described as follows:

$${}^G\mathbf{P}_{RR}(k) = \nabla \mathbf{R}_S^G(k) \begin{bmatrix} {}^G\mathbf{P}_{ss}(k) \\ {}^s\mathbf{P}_{RR}(k) \end{bmatrix} (\nabla \mathbf{R}_S^G(k))^T, \quad (25)$$

where $\nabla \mathbf{R}_S^G(k)$ is

$$\nabla \mathbf{R}_S^G(k) = \frac{\partial \mathbf{R}_S^G(k)}{\partial ({}^G\hat{\mathbf{X}}_S(k), {}^s\hat{\mathbf{X}}_R(k))}. \quad (26)$$

3.5. The Constraints for Local Feature Point Estimation. The local pose estimation of the Mars rover is independent of the global coordinates in the local submap algorithm. The estimation of the feature points in the global coordinates cannot be applied to the update process for the local submap without the connection between the coordinates. The feature points in the global coordinates and the local coordinates are independent, which results in repeated calculations when the feature points in the local coordinates are integrated into the global coordinates. The estimation of the feature points must be able to apply in the global estimation process.

In the general process, the consistent estimation of the feature points state can be obtained by the known constraints between the common state. The independent estimation of the feature points can be integrated into the global coordinate; meanwhile, a simple and consistent global estimation is generated by using the known connection between the state variables.

The projection is defined as follows:

$$\mathbf{C}(\hat{\mathbf{X}}(k)) = \mathbf{b}. \quad (27)$$

The constraint can be approximated estimation by EKF. The result can be described as follows:

$$\begin{aligned} \hat{\mathbf{X}}_c(k) &= \hat{\mathbf{X}}(k) + \mathbf{W}_c(k)(\mathbf{b} - \mathbf{C}(k)\hat{\mathbf{X}}(k)), \\ \mathbf{P}_c(k) &= \mathbf{P}(k) + \mathbf{W}_c(k)\mathbf{S}_c(k)\mathbf{W}_c^T(k), \end{aligned} \quad (28)$$

where

$$\begin{aligned} \mathbf{W}_c(k) &= \mathbf{P}(k)\nabla \mathbf{C}(k)\mathbf{S}_c^{-1}(k), \\ \mathbf{S}_c(k) &= \nabla \mathbf{C}(k)\mathbf{P}(k)\nabla \mathbf{C}^T(k). \end{aligned} \quad (29)$$

The system state variables are shown above, then the constraints must satisfy the condition, which is the constraints for local feature points integrating into the global map:

$${}^G\mathbf{X}_i(k) - ({}^G\mathbf{X}_S(k) \oplus {}^s\mathbf{X}_i(k)) = 0. \quad (30)$$

The constraints can be expressed as follows:

$$\mathbf{C}(k) \begin{bmatrix} {}^G\mathbf{X}_S(k) \\ {}^G\mathbf{X}_i(k) \\ {}^s\mathbf{X}_R(k) \\ {}^s\mathbf{X}_i(k) \end{bmatrix} = 0, \quad (31)$$

where

$$\mathbf{C}(k) = [-\mathbf{I} \quad \mathbf{I} \quad 0 \quad -\mathbf{I}], \quad (32)$$

$\mathbf{b} = 0$, and the constraint filtering can be applied to the conversion process to the global map for the local feature points.

4. Simulation Results

The simulation results are analyzed in this section. We can obtain the Mars terrain as the method mentioned in the paper [16]. We hypothesize that the real trajectory of the Mars rover is a circle. The 3D terrain and Mars rover's trajectory are shown in Figure 5. The following simulation results are mainly divided into two parts. The results shown from Figures 6–8 are simulated by standard EKF-SLAM algorithm; meanwhile, the results shown from Figures 9–12 are simulated by the new algorithm which is presented in this paper. The total number of the feature points considered in the state variables of EKF-SLAM and the new algorithm are compared in Figure 13. Since the computational complexity is proportional to the quadratic of the number of the feature points contained in a single filtering process, the main difference between the two algorithms can be shown in Figure 13. Obviously, the number of the feature points used in the new algorithm is much less than which are used in standard EKF-SLAM algorithm. The lower computational complexity is obtained by the SLAM algorithm with local submaps (Table 1).

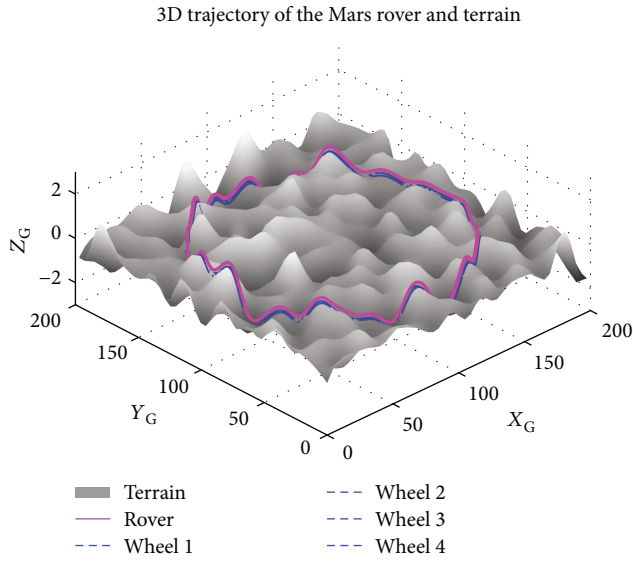


FIGURE 5: 3D terrain and Mars rover's trajectory 200 landmarks (asterisks) are distributed in the map. The simulation parameters are shown in Table 1.

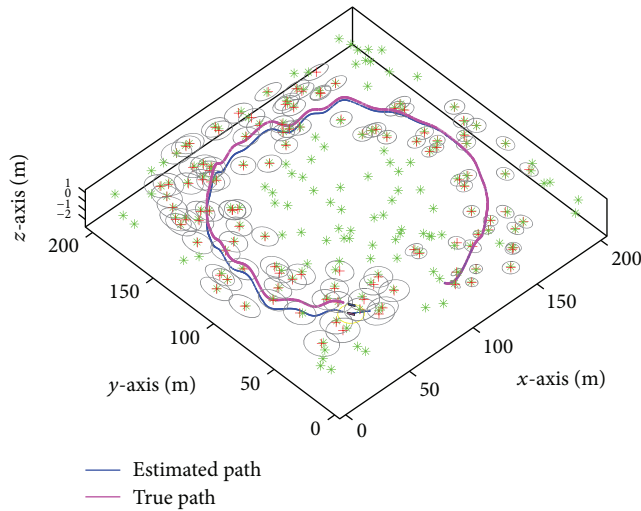


FIGURE 6: The true trajectory and estimated trajectory of the Mars rover by standard EKF-SLAM.

The true trajectory and estimated trajectory of the Mars rover by standard EKF-SLAM are shown in Figure 6. The pink solid line represents the true trajectory and the blue dashed line represents the estimated trajectory of the Mars rover. Meanwhile, the green asterisks represent land marks. The red asterisks represent the observed feature points. The position error is less than 1 m, and the attitude error is less than 0.5° , which can be seen from Figures 7 and 8.

In the following section, an improved EKF-SLAM approach with submaps is analyzed. The true trajectory and estimated trajectory of the Mars rover by improved EKF-SLAM algorithm with submaps are shown in Figure 9. The position error and attitude error can be seen in Figures 10 and 11 (1 m, 0.5°), which keep the same accuracy and error as the previous standard EKF-SLAM

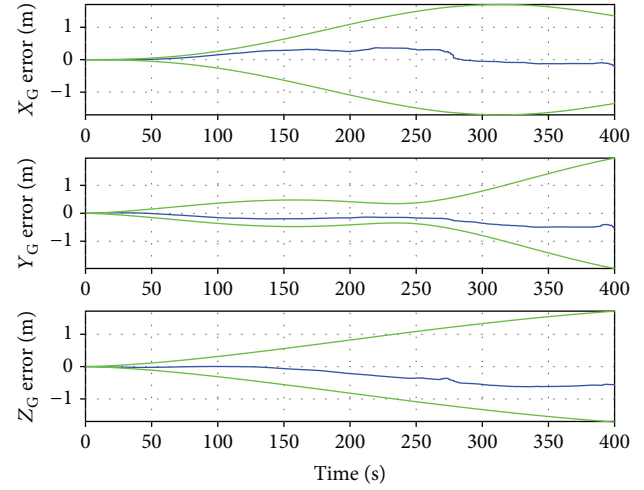


FIGURE 7: The position error curve of the Mars rover.

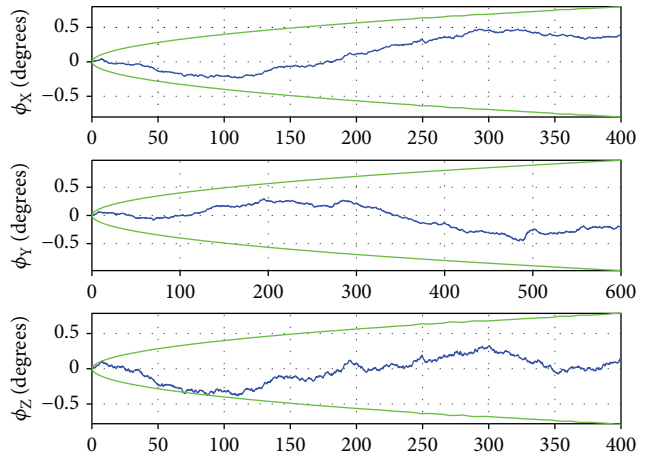


FIGURE 8: The attitude error of the Mars rover.

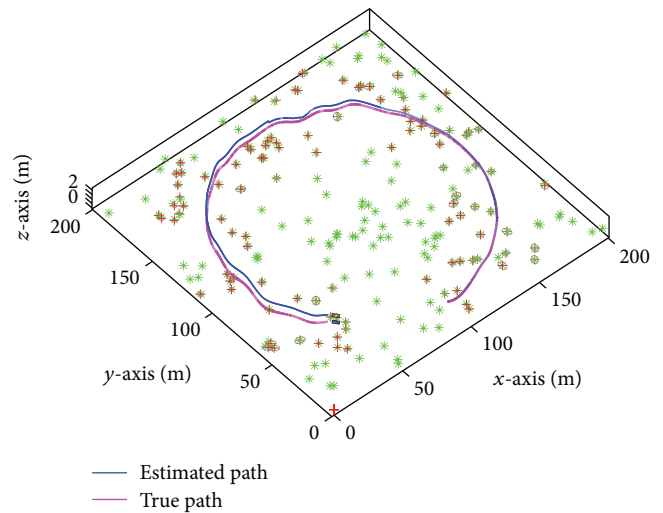


FIGURE 9: The true trajectory and estimated trajectory of the Mars rover by EKF-SLAM with submaps.

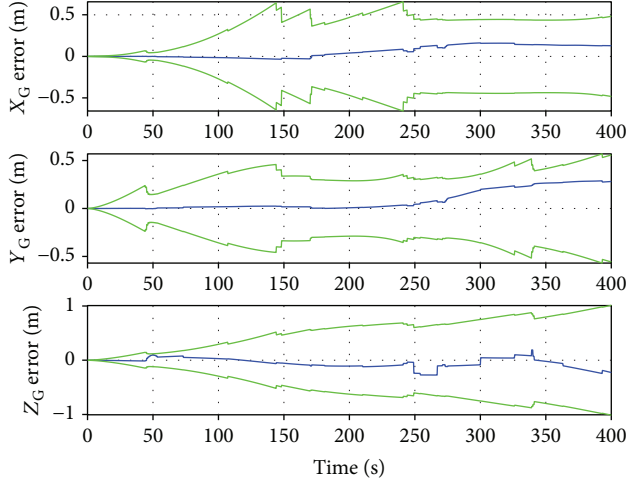


FIGURE 10: The pose error curve of the Mars rover.

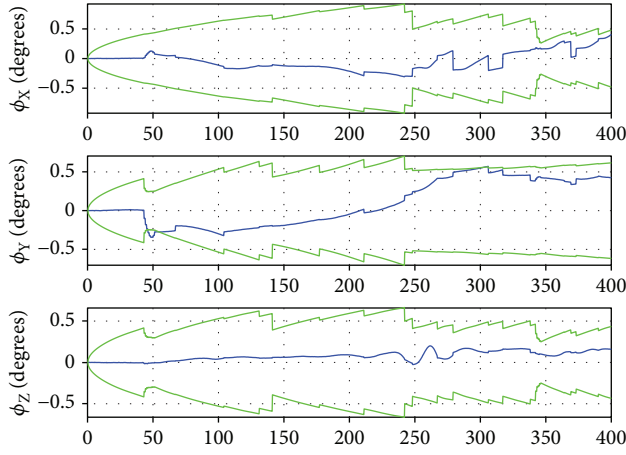


FIGURE 11: The attitude error of the Mars rover.

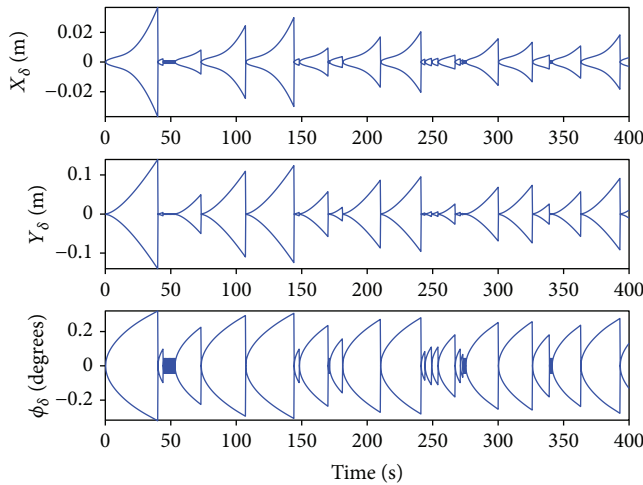


FIGURE 12: The standard deviation curve of the local position and azimuth deviation of the Mars rover.

algorithm. However, the total number of the feature points considered in the state variables of the improved algorithm with submaps is much smaller than that of the standard

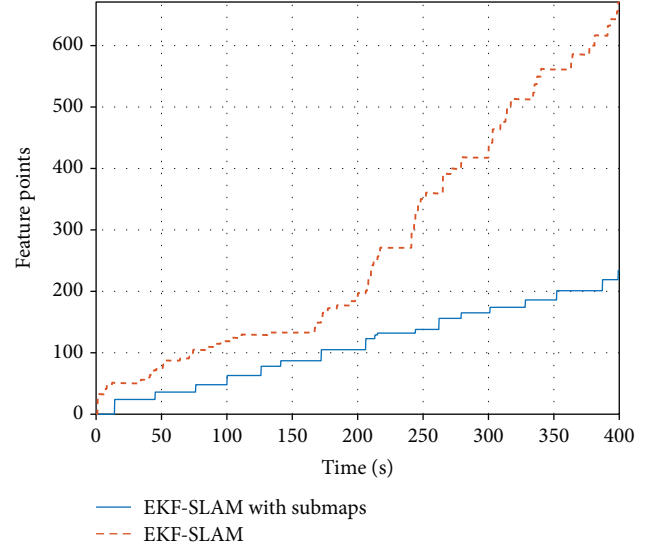


FIGURE 13: The total number of the feature points considered in the state variables of EKF-SLAM and EKF-SLAM with local submaps.

TABLE 1: The simulation parameters.

Name	Value
Sampling period	0.2 s
The number of feature points in the map	200
The radius of the wheel	0.3 m
The initial position of the rover	[100, 20, -0.3174]
The SD of RN of the accelerometer	2.0×10^{-3} m/s
The SD of RN of the gyroscope	0.02°/s
The SD of RN of the measuring length	1.0 m
The SD of RN of the measuring azimuth	3°
The SD of RN of the control stick	3°
Sensor frequency of the IMU	20 Hz
Sensor frequency of the camera	0.5 Hz
Simulation time	400 s

Note: SD denotes standard deviation; RN denotes random noise.

EKF-SLAM algorithm. As we know, the computational complexity is proportional to the quadratic of the number of the feature points contained in a single filtering process. Therefore, the computational complexity of the improved EKF-SLAM algorithm with submaps has decreased significantly compared with the standard EKF-SLAM algorithm. The standard deviation curve of the local position and azimuth deviation of the Mars rover is given in Figure 12. The total number of the feature points considered in the state variables of the standard EKF-SLAM and SLAM with submaps are shown in Figure 13. The computational complexity is proportional to the square of the number of the feature points contained in the state variables, and the total computational complexity of EKF-SLAM algorithm is proportional to the cube of the final number of feature points in the state variables. Therefore, the computational complexity is reduced sharply in SLAM with submaps. Some

TABLE 2: Some comparison results of the two algorithms.

	Standard EKF-SLAM	EKF-SLAM with submaps
The total number of the feature points considered in the state variables	700	440
Computational complexity	$\propto(700)^3$	$\propto(440)^3$
The triaxial position error of Mars rover	≤ 1 m	≤ 0.5 m
The triaxial angle error of Mars rover	$\leq 0.5^\circ$	$\leq 0.5^\circ$

comparison results of two algorithms are listed in Table 2, from which we can see the difference of the two algorithms clearly.

5. Conclusion

In the paper, we proposed an improved EKF-SLAM algorithm with submaps for Mars surface exploration. The presented method could reduce the computational complexity to a great extent. The error of the position and attitude could be controlled in an acceptable ranges (1 m, 0.5°) for both two algorithms, which can be seen in the simulation results. The position error of the EKF-SLAM with submaps is smaller than that of the standard EKF-SLAM. Meanwhile, the computational complexity is lower by the improved algorithm, which is more suitable for the complex terrain of Mars.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

The paper is completed with the help of Professor Franco Bernelli-Zazzera and Professor Mauro Massari of the Aerospace Engineering Department of Politecnico di Milano.

References

- [1] B. Ayhan, M. Dao, C. Kwan, H. M. Chen, J. F. Bell, and R. Kidd, "A novel utilization of image registration techniques to process mastcam images in mars rover with applications to image fusion, pixel clustering, and anomaly detection," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 10, pp. 4553–4564, 2017.
- [2] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johnson, A. Ansar, and L. Matthies, "Vision-aided inertial navigation for spacecraft entry, descent, and landing," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 264–280, 2009.
- [3] J. Matthaei, T. Krüger, S. Nowak, and U. Bestmann, "Swarm exploration of unknown areas on Mars using SLAM," in *International Micro Air Vehicle Conference and Flight Competition*, Toulouse, France, 2013.
- [4] V. Imani, K. Haataja, and P. Toivanen, "Three main paradigms of simultaneous localization and mapping (SLAM) problem," in *Proceedings Volume 10696, Tenth International Conference on Machine Vision (ICMV 2017)*, Vienna, Austria, April 2018.
- [5] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.
- [6] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part II," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 3, pp. 108–117, 2006.
- [7] J. A. Castellanos, R. Martinez-Cantin, J. D. Tardós, and J. Neira, "Robocentric map joining: improving the consistency of EKF-SLAM," *Robotics and Autonomous Systems*, vol. 55, no. 1, pp. 21–29, 2007.
- [8] R. Martinez-Cantin and J. A. Castellanos, "Bounding Uncertainty in EKF-SLAM: the Robocentric Local Approach," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*, pp. 430–435, Orlando, FL, USA, May 2006.
- [9] L. M. Paz, J. D. Tardós, and J. Neira, "Divide and conquer: EKF SLAM in $O(n)$," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1107–1120, 2008.
- [10] S. B. Samsuri, H. Zamzuri, M. A. A. Rahman, S. A. Mazlan, and A. H. A. Rahman, "Computational cost analysis of extended kalman filter in simultaneous localization and mapping (EKF-SLAM) problem for autonomous vehicle," *ARPJN Journal of Engineering and Applied Sciences*, vol. 10, no. 17, pp. 153–158, 2015.
- [11] C. C. Hsu, C. K. Yang, Y. H. Chien, Y. T. Wang, W. Y. Wang, and C. H. Chien, "Computationally efficient algorithm for vision-based simultaneous localization and mapping of mobile robots," *Engineering Computations*, vol. 34, no. 4, pp. 1217–1239, 2017.
- [12] Y. Jeng, A. Jain, and J. Balaram, *ROAMS: Rover analysis, modeling and simulation software*, 1999.
- [13] M. Yu, H. Cui, and Y. Tian, "A new approach based on crater detection and matching for visual navigation in planetary landing," *Advances in Space Research*, vol. 53, no. 12, pp. 1810–1821, 2014.
- [14] Y. Y. Dong, *Simultaneous Localization and Mapping for Mars Rovers, [M.S. thesis]*, Harbin Institute of Technology, 2015.
- [15] M. Yu, "Research on autonomous visual navigation method for planetary landing and exploration mission," *Doctoral dissertation*, Harbin Institute of Technology, 2016.
- [16] M. Tarokh and G. J. McDermott, "Kinematics modeling and analyses of articulated rovers," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 539–553, 2005.

